AMENDMENTS TO THE CLAIMS

1. (Currently Amended) A system comprising:

a computer system having a processor coupled with a memory, the computer system further including an application server, the application server ~~including~~ to generate a unified logging and tracing system ~~including~~ having a logging system to generate log messages, and a tracing system to generate trace messages, ~~the unified logging and tracing system~~ wherein the application server is further to cross-reference the trace messages and the log messages or generate language-independent messages for the log messages, wherein the unified logging and tracing system is compatible with multiple output formats and is controllable such that setting hierarchies of the unified logging and tracing system are reusable, the unified logging and tracing system having

a log manager to manage one or more log controllers to receive one or more messages from an application wherein each of the one or more log controllers is a class that includes one or more subclasses or modules selected from a group comprising a category to generate the log messages and a location to generate the trace messages, ~~wherein the generating of the trace messages includes reconstructing a control flow of the application while executing,~~ wherein the trace messages are emitted to the location, the location including an area of program code; and

~~a log manager coupled to the one or more log controllers to manage the one or more log controllers;~~

~~one or more logs to which the received messages are forwarded;~~

~~a formatter coupled to the one or more logs, the formatter to format each of the one or more messages prior to publication of the one or more messages, wherein each of the one or more messages is associated with a log record, wherein the log record includes severity information, the severity information having one or more of debug, path, info, warning, error, fatal, and none; and~~

a viewer~~ coupled to the formatter~~ in communication with the processor of the computer system, the viewer to display~~ the formatted one or more messages~~ the logging messages and the tracing messages as generated by the unified logging and tracing system.

2. (Original) The system of claim 1, wherein the log manager is part of a kernel of a Java 2 Enterprise Edition (J2EE) Engine.

3. (Original) The system of claim 1, wherein the log manager configures a plurality of logging routines for a kernel.

4. (Original) The system of claim 1, wherein the log manager provides support for storing system critical logs in a database.

5. (Cancelled)

6. (Currently Amended) The system of claim 1, wherein the unified logging and tracing system comprises a formatter to format each of the logging and tracing messages prior to publication of the logging and tracking messages, wherein each of the logging messages is associated with a log record, wherein the log record includes severity information, the severity information having one or more of debug, path, info, warning, error, fatal, and none, wherein the formatter includes

one or more subclasses or modules selected from a group comprising a list formatter, a trace formatter, and an Extensible Markup Language (XML) formatter.

7. (Currently Amended) The system of claim ~~1~~ 6, wherein the formatter is associated with one or more logs, wherein each of the one or more logs ~~include~~ includes one or more subclasses or modules selected from a group comprising a stream log, a file log, and a console log.

8. (Previously Presented) The system of claim 7, wherein one or more of the file log and the console log include subclasses of the stream log.

Claims 9-12 (Cancelled)

13. (Currently Amended) A method comprising:

~~executing~~ generating a unified logging and tracing system ~~including~~ having a logging system to generate log messages, and a tracing system to generate trace messages, and cross-referencing the trace messages and the log messages or generating language-independent messages for the log messages, wherein the unified logging and tracing system is compatible with multiple output formats and is controllable such that setting hierarchies of the unified logging and tracing system are reusable, the unified logging and tracing system; ~~wherein the~~

executing ~~of~~ the unified logging and tracing system, wherein executing ~~further~~ includes receiving one or more messages from an application via one or more log controllers being managed by a log manager, wherein each of the one or more log controllers is a class that includes one or more subclasses or modules selected from a group comprising a category to generate the log messages and a location to generate the trace messages, ~~wherein the generating of the trace messages includes reconstructing a control flow of the application while executing,~~ wherein the trace messages are emitted to the location, the location including an area of program code; and

~~managing the one or more log controllers via a log manager coupled to the one ore more log controllers;~~

~~forwarding the received messages to one or more logs;~~

~~formatting each of the one or more messages prior to publication of the one or more messages via a formatter coupled to the one or more logs, wherein each of the one or more messages is associated with a log record, wherein the log record includes severity information, the severity information having one or more of debug, path, info, warning, error, fatal, and none; and~~

displaying ~~the formatted one or more messages via a viewer coupled to the formatter~~the logging messages and the tracing messages as generated by the unified logging and tracing system.

14. (Previously Presented) The method of claim 13, wherein the log manager configures a plurality of logging routines for a kernel.

15. (Previously Presented) The method of claim 13, wherein the log manager provides support for storing system critical logs in a database.

16. (Currently Amended) The method of claim 13, wherein the unified logging and tracing system comprises a formatter to format each of the logging and tracing messages prior to publication of the logging and tracking messages, wherein each of the logging messages is associated with a log record, wherein the log record

includes severity information, the severity information having one or more of debug, path, info, warning, error, fatal, and none, wherein the formatter includes one or more subclasses or modules selected from a group comprising a list formatter, a trace formatter, and an Extensible Markup Language (XML) formatter.

17. (Currently Amended) The method of claim ~~13~~ 16, wherein the formatter is associated with one or more logs, wherein each of the one or more logs ~~include~~ includes one or more subclasses or modules selected from a group comprising a stream log, a file log, and a console log.

18. (Previously Presented) The method of claim 17, wherein one or more of the file log and the console log include subclasses of the stream log.

19. (Currently Amended) A ~~tangible~~ machine-readable storage medium having instructions which, when executed, cause a machine to:

~~executing~~ generate a unified logging and tracing system ~~including~~ having a

logging system to generate log messages, and a tracing system to generate trace messages, and cross-referencing the trace messages and the log messages or generating language-independent messages for the log messages, wherein the unified logging and tracing system is compatible with multiple output formats and is controllable such that setting hierarchies of the unified logging and tracing system are reusable, the unified logging and tracing system; ~~wherein the~~

executing of the unified logging and tracing system, wherein executing ~~further~~

includes receive one or more messages from an application via one or more log controllers being managed by a log manager, wherein each of the one or more log controllers is a Java class that includes one or more subclasses or modules selected from a group comprising a category to generate the log messages and a location to generate the trace messages, ~~wherein the generating of the trace messages includes reconstructing a control flow of the application while executing,~~ wherein the trace messages are emitted to the location, the location including an area of prorgam code; and

~~manage the one or more log controllers via a log manager coupled to the one ore more log controllers;~~

~~forward the received messages to one or more logs;~~

~~format each of the one or more messages prior to publication of the one or more messages via a formatter coupled to the one or more logs, wherein each of the one or more messages is associated with a log record, wherein the log record includes severity information, the severity information having one or more of debug, path, info, warning, error, fatal, and none; and~~

display ~~the formatted one or more messages via a viewer coupled to the formatter~~the logging messages and the tracing messages as generated by the unified logging and tracing system.

20. (Currently Amended) The ~~tangible~~ machine-readable storage medium of claim 19, wherein the log manager provides support for storing system critical logs in a database.

21. (Currently Amended) The ~~tangible~~ machine-readable storage medium of claim 19, wherein the unified logging and tracing system comprises a formatter to format each of the logging and tracing messages prior to publication of the logging and

tracking messages, wherein each of the logging messages is associated with a log record, wherein the log record includes severity information, the severity information having one or more of debug, path, info, warning, error, fatal, and none, wherein the formatter includes one or more subclasses or modules selected from a group comprising a list formatter, a trace formatter, and an Extensible Markup Language (XML) formatter.

22. (Currently Amended) The ~~tangible~~ machine-readable storage medium of claim ~~19~~ 21, wherein each of the one or more logs include one or more subclasses or modules selected from a group comprising a stream log, a file log, and a console log.

23. (Currently Amended) The ~~tangible~~ machine-readable storage medium of claim 19, wherein one or more of the file log and the console log include subclasses of the stream log.